HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400

PATENT APPLICATION

ATTORNEY DOCKET NO. 10992668-1

IN THE
UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Slawomir Ilnicki et al.

Confirmation No.:

Application No.: 09/952,322

Examiner: Christopher Revak

Filing Date: 6/13/00

Group Art Unit: 2131

Title: Secure Data Transfer Method and System

Mail Stop Appeal Brief-Patents
Commissioner For Patents
PO Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL OF APPEAL BRIEF

Sir:

Transmitted herewith is the Appeal Brief in this application with respect to the Notice of Appeal filed on 11-12-2004.

The fee for filing this Appeal Brief is (37 CFR 1.17(c)) $500.00.

(complete (a) or (b) as applicable)

The proceedings herein are for a patent application and the provisions of 37 CFR 1.136(a) apply.

( ) (a) Applicant petitions for an extension of time under 37 CFR 1.136 (fees: 37 CFR 1.17(a)-(d) for the total number of months checked below:

( ) one month	$120.00
( ) two months	$450.00
( ) three months	$1020.00
( ) four months	$1590.00

( ) The extension fee has already been filled in this application.

( ) (b) Applicant believes that no extension of time is required. However, this conditional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition and fee for extension of time.

Please charge to Deposit Account **08-2025** the sum of **$500.00**. At any time during the pendency of this application, please charge any fees required or credit any over payment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.16 through 1.21 inclusive, and any other sections in Title 37 of the Code of Federal Regulations that may regulate fees. A duplicate copy of this sheet is enclosed.

(X) I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, Alexandria, VA 22313-1450. Date of Deposit: 12/27/2004

OR

( ) I hereby certify that this paper is being transmitted to the Patent and Trademark Office facsimile number _____ on _____

Number of pages:

Typed Name: Be Henry

Signature: B. Henry

Rev 12/04 (Aplbrief)

Respectfully submitted,

Slawomir Ilnicki et al.

By _____

Philip Lyren

Attorney/Agent for Applicant(s)
Reg. No. **40,709**

Date: **12/27/04**

Telephone No.: **281 514 8232**

01/05/2005 ZJUHAR1 00000108 082025 09952322
01 FC:1402 500.00 DA

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Appellants:	Slawomir Ilnicki et al.	Examiner: Christopher A. Revak
Serial No.:	09/592,322	Group Art Unit: 2131
Filed:	June 13, 2000	Docket No.: 10992668-1
Title:	Secure Data Transfer Method and System	

APPEAL BRIEF UNDER 37 C.F.R. § 41.37

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This Appeal Brief is filed in response to the Final Office Action mailed July 15, 2004 and the Notice of Appeal filed on November 12, 2004.

AUTHORIZATION TO DEBIT ACCOUNT

It is believed that no extensions of time or fees are required, beyond those that may otherwise be provided for in documents accompanying this paper. However, in the event that additional extensions of time are necessary to allow consideration of this paper, such extensions are hereby petitioned under 37 C.F.R. § 1.136(a), and any fees required (including fees for net addition of claims) are hereby authorized to be charged to Hewlett-Packard Development Company's deposit account no. 08-2025.

REAL PARTY IN INTEREST

The real party-in-interest is the assignee, Hewlett-Packard Company, a Delaware corporation, having its principal place of business in Palo Alto, California.

RELATED APPEALS AND INTERFERENCES

There are no known related appeals or interferences known to appellant, the appellant's legal representative, or assignee that will directly affect or be directly affected by or have a bearing on the Appeal Board's decision in the pending appeal.

STATUS OF CLAIMS

Claims 1 - 26 stand finally rejected. No claims have been allowed. The final rejection of claims 1 - 26 is appealed.

STATUS OF AMENDMENTS

In the Final Office Action, claim 23 was rejected under 35 USC § 112, second paragraph, as being indefinite. In response to the Final Office Action, Appellants attempted to cancel claim 23. In the Advisory (date mailed: 11/04/2004), the Examiner refused to enter cancellation of claim 23.

The Claims Appendix corresponds to the claims submitted in the Response (date mailed: 09/09/2004) to the Final Office Action. Although claim 23 is shown as "canceled," this amendment was not entered.

SUMMARY OF CLAIMED SUBJECT MATTER

The summary is set forth in five exemplary embodiments that correspond to independent claims 1, 11, 15, 17, and 22. Discussions about elements and recitations of these claims can be found at least at the cited locations in the specification and drawings.

Claim 1

A method for securely transferring data between an agent and an application server through a non-secure node (Fig. 3, block 314; Fig. 5; Page 8, lines 10+) comprising:

(a) establishing a session key between the agent and the application server by utilizing a public key of the application server (Fig. 3, block 304; Fig. 4; Page 9, lines 21-25); wherein the public key of the application server is embedded in the agent to enable the agent to derive the session key (Fig. 4, #408; Page 10, lines 7-31); and

(b) establishing an end-to-end secure connection between the agent and the application server by using the session key and by establishing a communication link between the application server and the non -secure node by using a relay module (Fig. 3, block 308; Fig. 6; Fig. 8; corresponding description to figures).

Claim 11

The method of securely transferring data between an application server and an agent of the application server through a non-secure environment having a web-server and the agent (Fig. 3, block 314; Fig. 5; Page 8, lines 10+), the method comprising:

a) a user accessing the web-server to download the agent therefrom (Fig. 4, #400; Page 10, lines 1-5); wherein the agent includes a public key of the application server (Page 10, lines 1-13);

b) the agent deriving a shared session key with the application server by using the public key of the application server, the shared session key for use in encrypting and decrypting data to be transferred between the agent and the application server (Fig. 3, block 304; Fig. 4, #408; Page 10, lines 14-31);

c) the application server establishing a connection to the web-server (Fig. 3, block 308; Fig. 8); and

d) the agent contacting the web server by using a first protocol to send data encrypted by the session key to the application server over the connection between the web-server and the application server (Fig. 3, block 314; Fig. 7; Fig. 8; corresponding description to figures).

Claim 15

A secure data transfer system for connecting a non-secure node to an application server behind a firewall (Fig. 5, #500; Page 8, lines 10-15) comprising:

a) a web-server (Fig. 5, # 540) in the non-secure node (Fig. 5, # 530; Page 8, lines 22-24);

b) a relay (Fig. 5, # 560) in the non-secure node that is dynamically instantiated by the application server (Fig. 5, # 524), the relay being configured by the application server to have a first port (Fig. 5, # 561) for listening for a connection from the application server (Fig. 6; Fig. 8; Page 8, line 25 – page 9, line 8);

wherein the application server connects to the relay on the first port and reads data from the first port (Fig. 6; Fig. 8; Page 9, lines 1-8).

Claim 17

A secure data transfer system for establishing an end-to-end secure connection between an agent and an application server behind a firewall through a non-secure node (Fig. 5, # 500; Page 8, lines 10-15) comprising:

a) a web-server (Fig. 5, # 540) residing in the non-secure node (Fig. 5, # 530), the web-server having the agent that includes a public key of the application server (Fig. 5, # 524; Page 8, lines 10-24);

b) a browser (Fig. 5, # 510) in communication with the web-server for downloading the agent from the web-server (Page 8, lines 19-21);

c) a secure transfer module (Fig. 5, # 548) residing in the non-secure node (Fig. 5, # 530; Page 8, lines 25-30); and

d) an application server (Fig. 5, # 524) in a secure zone (Fig. 5, # 520) for initiating a connection to the web-server via the secure transfer module (Fig. 3; Page 8, lines 25-30; Fig. 3).

Claim 22

A method, comprising:

embedding in code of an agent a public key of an application server that is behind a firewall (Fig. 4, #400, Page 10, lines 1-6; Fig. 5);

downloading the code of the agent and the public key into a browser (Fig. 4, #400; Page 10, lines 1-6; Fig. 5);

verifying the agent to authenticate the public key of the application server (Fig. 4, # 404; Page 10 line 7 – page 11, line 8);

establishing a communication link between the application server and a relay module that is in a non-secure environment and between the browser and the relay module (Fig. 3, block 308; Fig. 5; Fig. 6; Page 9, lines 25-28); and

securely transferring data from the browser through the relay module to the application server without requiring a trusted intermediate party (Fig. 3, block 314; Fig. 5; Fig. 8; Page 9, lines 28-29).

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

I. Claims Rejection - 35 USC § 112

Claim 23 is rejected under 35 USC § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

II. Claims Rejection (Claims 1-5, 11, 14-22, 25-26) – 35 USC § 102(e)

Claims 1-5, 11, 14-22, and 25-26 are rejected under 35 USC § 102(e) as being anticipated by Baker et al. (USPN 6,611,498, hereinafter "Baker").

III. Claims Rejection (Claims 6-8) – 35 USC § 103(a)

Claims 6-8 are rejected under 35 USC §103(a) as being unpatentable over Baker in view of Curry et al. (USPN 6,237,095, hereinafter Curry).

IV. Claims Rejection (Claims 9-10) – 35 USC § 103(a)

Claims 9-10 are rejected under 35 USC §103(a) as being unpatentable over Baker in view of Boyle et al. (USPN 6,119,167, hereinafter Boyle).

V. Claims Rejection (Claims 12 and 16) – 35 USC § 103(a)

Claim 12 is rejected under 35 USC §103(a) as being unpatentable over Baker in view of Bradley et al. (USPN 6,584,507, hereinafter Bradley).

VI. Claim Rejection (Claim 13) – 35 USC § 103(a)

Claim 13 is rejected under 35 USC §103(a) as being unpatentable over Baker in view of Bradley and Curry

VII. Claim Rejection (Claim 24) – 35 USC § 103(a)

Claim 24 is rejected under 35 USC §103(a) as being unpatentable over Baker in view of Curry.

ARGUMENT

I. Claims Rejection - 35 USC § 112

The Final Office Action (dated 07/15/2004) rejected claim 23 under 35 USC § 112, second paragraph, as being indefinite. This claim, as submitted, is reproduced below:

> 23. (new) The method of claim 22 wherein the trusted intermediate party is selected from the group consisting of a trusted node, a trusted server, and a secure channel.

In response to the Final Office Action, Appellants attempted to cancel claim 23. In the Advisory (date mailed: 11/04/2004), the Examiner refused to enter cancellation of claim 23.

Appellants contend that the refusal to cancel claim 23 is improper. 37 CFR § 1.116(b)(1) and (2) state:

> (1) An amendment may be made canceling claims or complying with any requirement of form expressly set forth in a previous Office action.
> (2) An amendment presenting rejected claims in better form for consideration on appeal may be admitted.

Cancellation of claim 23 does not affect the scope of any other pending claim. In fact, cancellation of this claim was made merely to place the application in a better form for appeal.

Nonetheless, the Examiner has refused to cancel claim 23. Appellants respectfully request reversal of the decision of the Examiner.

II. Claims Rejection (Claims 1-5, 11, 14-22, 25-26) – 35 USC § 102(e)

Claims 1-5, 11, 14-22, and 25-26 are rejected under 35 USC § 102(e) as being anticipated by Baker et al. (USPN 6,611,498, hereinafter "Baker").

Appellants respectfully contest the rejection of all of these claims. Further, each of the independent claims (1, 11, 15, 17, and 22) is separately argued below with a separate sub-heading.

Sub-Heading I: Claim 1 not anticipated by Baker

Claim 1 is rejected under 35 U.S.C. §102 as allegedly anticipated by Baker. Claim 1 recites numerous limitations that are not taught or suggested in Baker; examples are discussed below. For convenience, claim 1 is reproduced (emphasis added):

> A method for securely transferring data between **an agent** and an application server through a non-secure node comprising:
>
> (a) establishing a session key between **the agent** and the application server by utilizing a public key of the application server; **wherein the public key of the application server is embedded in the agent to enable the agent to derive the session key**; and
>
> (b) establishing an end-to-end secure connection between **the agent** and the application server by using the session key and by establishing a communication link between the application server and the non-secure node by using a relay module.

On numerous occasions, claim 1 recites recitations pertaining to an **agent**. The Office Action cites Baker (at Col. 5, lines 60-61) for teaching this limitation. This section of Baker teaches a client tier of software on a customer workstation that has "one or more downloadable application objects directed to front-end business logic." Baker, thus, teaches downloading an application object into a workstation. Baker does not teach or suggest that this application object is utilized as an "agent" per the plain meaning of this term as understood in the art.

According to MPEP § 2111.01, the words of a claim must be given their "plain meaning." Webopedia is an online dictionary for computer and internet technology definitions. Per Webopedia (see www.webopedia.com), an agent is defined as: "A program that performs some information gathering or processing task in the background.

Typically, an agent is given a very small and well-defined task." Appellants submit that Baker does not teach or suggest that the downloaded application objects are utilized as an "agent" per the plain meaning of this term.

Appellants acknowledge that claims must be given their broadest interpretation during patent examination. However, this interpretation must be a "**reasonable interpretation consistent with the specification**" (see MPEP 2111: emphasis added). Appellants specification repeatedly uses the term "agent" in a manner consistent with the plain meaning of this term. Appellants respectfully ask the Board of Appeals to read Appellants' Background of the Invention for a discussion of exemplary agents.

Claim 1 recites numerous recitations that include the word "agent." Baker fails to teach or suggest each of these recitations. For example, claim 1 recites:

1) securely transferring data between **an agent** and an application server,
2) establishing a session key between **the agent** and the application server,
3) wherein the public key of the application server is embedded in **the agent**,
4) establishing an end-to-end secure connection between **the agent** and the application server.

Nowhere does Baker teach or suggest using an agent as recited in claim 1. For at least these reasons, Appellants respectfully request withdrawal of the rejection.

As an additional example, claim 1 recites that "the public key of the application server is embedded in the agent to enable the agent to derive the session key." This limitation is not taught in Baker.

First, Baker (Col 9, lines 10-12) teaches a server to generate a "cookie" that is sent to the client. A "cookie" is **not** a public key. These two terms have entirely different meanings to one of ordinary skill in the art. According to MPEP § 2111.01, the words of a claim must be given their "plain meaning." Per Webopedia (see www.webopedia.com), a cookie is defined as: "A message given to a Web browser by a Web server. The browser stores the message in a text file. The message is then sent back to the server each time the browser requests a page from the server." By contrast, a public key is defined as: "A cryptographic system that uses two keys -- a public key known to everyone and a

private or secret key known only to the recipient of the message." Thus, Baker does not teach "the public key of the application server is embedded in the agent."

Second, Baker teaches a "cookie" that is sent to the client and then returned to the server. Specifically, Baker teaches:

> The preferred embodiment further associates a given HTTPS request with a logical session which is initiated and tracked by a "cookie jar server" 28 to generate a "cookie" which is a unique server-generated key that is sent to the client along with each reply to a HTTPS request. The client holds the cookie and returns it to the server as part of each subsequent HTTPS request. (Col. 9, lines 7-13).

Claim 1 recites that the public key of the application server is embedded in the agent. This limitation is not taught or suggested in Baker. Baker teaches a server that sends a cookie to a client. Sending a cookie to a client does not teach a public key embedded in **an agent**.

Thirdly, claim 1 also recites a public key of the application server embedded in the agent **to enable the agent to derive the session key**. In Baker, the cookie jar server generates a cookie and sends it to the client. The client then holds the cookie and sends it back to the server. The "cookie" is not embedded in an agent to enable the cookie to derive a session key.

For at least these reasons, Appellants respectfully request withdrawal of the rejection.

Thus, the cited art does not teach or suggest each and every limitation of claim 1. All dependent claims that depend from independent claim 1 inherit all limitations of the base claim. For at least the reasons given in connection with claim 1, the dependent claims are also allowable over Baker.

Sub-Heading II: Claim 11 not anticipated by Baker

Claim 11 is rejected under 35 U.S.C. §102 as allegedly anticipated by Baker. Claim 11 recites numerous limitations that are not taught or suggested in Baker;

examples are discussed below. For convenience, claim 11 is reproduced (emphasis added):

> The method of securely transferring data between an application server and **an agent** of the application server through a non-secure environment having a web-server and **the agent**, the method comprising:
>
> a) **a user accessing the web -server to download the agent therefrom; wherein the agent includes a public key of the application server**;
>
> b) **the agent deriving a shared session key with the application server by using the public key of the application server**, the shared session key for use in encrypting and decrypting data to be transferred between **the agent** and the application server;
>
> c) the application server establishing a connection to the web-server; and
>
> d) **the agent** contacting the web server by using a first protocol to send data encrypted by the session key to the application server over the connection between the web-server and the application server.

On numerous occasions, claim 11 recites recitations pertaining to an **agent**. Baker does not teach the limitations of claim 11 regarding the agent. Appellants refer to Sub-Heading I above for a definition of the word "agent." Appellants provide a few additional examples to show that Baker does not teach or suggest the recitations of claim 11.

As one example, claim 11 recites "a user accessing the web-server to download the agent therefrom." The Office Action cites Baker (Col. 14, lines 7-10) for teaching this limitation. This section is reproduced:

> As shown in FIG. 7, the client desktop systems 630 with Internet connectivity have standard browsers executing Java applets, hereinafter referred to also as a client GUI application, downloaded from the web server 632.

This section of Baker teaches that a desktop can download a GUI application from a web server. This section does not teach a user accessing a web-server to download "an agent" from the web-server.

As another example, the claim recites that the agent includes a public key of the application server. The Office Action cites a section of Baker (Col. 9, lines 10-12) that discusses "cookies." Appellants respectfully assert that the Office Action is citing unrelated sections of Baker and pasting them together in an effort to teach the limitations of claim 11. In effect, the Office Action is constructing an improper "piecemeal examination" of the art and Appellants' claims. For at least this reason, Appellants respectfully request withdrawal of the rejection.

As noted, claim 11 recites that the agent includes a public key of the application server. This limitation is not disclosed in Baker. The Office Action cites Baker (Col. 11, lines 34-38) to teach utilizing a public key and then cites Baker (Col. 9, lines 10-12) to teach an agent including a public key. Appellants respectfully disagree. Baker (Col. 11, lines 34-38) teaches public key encryption, such as employed by a secure sockets layer (SSL). Baker (Col 9, lines 10-12) teaches a server to generate a "cookie" that is sent to the client. A "cookie" is **not** a public key. These two terms have entirely different meanings to one of ordinary skill in the art. Per Webopedia (see www.webopedia.com), a cookie is defined as: "A message given to a Web browser by a Web server. The browser stores the message in a text file. The message is then sent back to the server each time the browser requests a page from the server." By contrast, a public key is defined as: "A cryptographic system that uses two keys -- a public key known to everyone and a private or secret key known only to the recipient of the message." Thus, these sections taken together do not teach an agent that includes a public key of the application server.

As yet another example, claim 11 recites that the agent **derives** a shared session key with the application sever by using the public key of the application server. This limitation is not taught in Baker. The Office Action cites Baker (Col. 17, lines 7-11). Appellants reproduce the cited section of Baker:

> When a client logs onto the web server 632 and is authenticated, the client is provided a "session id" which is a unique server-generated key. The client holds this and

> returns it to the server as part of subsequent message transaction. (Col. 17, lines 6-10).

Note the difference between the recitations of claim 11 and the teachings of this section. Claim 11 recites that the agent **derives** the shared session key. The cited section of Baker teaches a client that is provided with a session id. In Baker, the client does not "derive" the session id.

Thus, the cited art does not teach or suggest each and every limitation of claim 11. All dependent claims that depend from independent claim 11 inherit all limitations of the base claim. For at least the reasons given in connection with claim 11, the dependent claims are also allowable over Baker.

Sub-Heading III: Claim 15 not anticipated by Baker

Claim 15 is rejected under 35 U.S.C. §102 as allegedly anticipated by Baker. Claim 15 recites numerous limitations that are not taught or suggested in Baker; examples are discussed below. For convenience, claim 15 is reproduced (emphasis added):

> A secure data transfer system for connecting a non-secure node to an application server behind a firewall comprising:
>
> a) a web-server in the non-secure node;
>
> b) a relay in the non-secure node that is dynamically instantiated by the application server, **the relay being configured by the application server to have a first port for listening for a connection from the application server**;
>
> wherein the application server connects to the relay on the first port and reads data from the first port.

Claim 15 recites numerous limitations that are not taught or suggested in Baker. For example, claim 15 recites that the relay is "configured by the application server." This limitation is not taught in Baker. Further, the claim recites that the relay is configured by the application server to have "a first port for listening for a connection from the application server." This limitation is not taught in Baker.

The Office Action cites Baker Col. 10, lines 23-34 and Col. 18, lines 13-25. These sections are individually addressed.

The first citation of Baker (i.e., Col. 10, lines 23-34) teaches a proxy server that "waits for requests from an application client running on the customer's workstation 10 and then services the request." (Col. 10, lines 25-27). This section does not teach a relay being configured by the application server to have a first port for listening for a connection from the application server.

The second citation of Baker (i.e., Col. 18, lines 13-25) teaches porting the proxy server over to the CMIDS (see Figs. 2 and 10). Nowhere does Baker teach a relay being configured by the application server to have a first port for listening for a connection from the application server.

Thus, the cited art does not teach or suggest each and every limitation of claim 15. All dependent claims that depend from independent claim 15 inherit all limitations of the base claim. For at least the reasons given in connection with claim 15, the dependent claims are also allowable over Baker.

Sub-Heading IV: Claim 17 not anticipated by Baker

Claim 17 is rejected under 35 U.S.C. §102 as allegedly anticipated by Baker. Claim 17 recites numerous limitations that are not taught or suggested in Baker; examples are discussed below. For convenience, claim 17 is reproduced (emphasis added):

> A secure data transfer system for establishing an end-to-end secure connection between **an agent** and an application server behind a firewall through a non-secure node comprising:
>
> a) a web-server residing in the non-secure node, **the web-server having the agent that includes a public key of the application server**;
>
> b) a browser in communication with the web-server for downloading **the agent** from the web-server;
>
> c) a secure transfer module residing in the non-secure node; and

d) an application server in a secure zone for initiating a connection to the web-server via the secure transfer module.

On numerous occasions, claim 17 recites recitations pertaining to an **agent**. Baker does not teach or suggest the limitations of claim 17 regarding the agent. Appellants provide a few examples to show that Baker does not teach or suggest the recitations in claim 17.

For example, the Office Action cites Baker (at Col. 5, lines 60-61) for teaching establishing an end-to-end secure connection between "an agent and an application server." This section of Baker teaches a client tier of software on a customer workstation that has "one or more downloadable application objects directed to front-end business logic." Baker, thus, teaches downloading an application object into a workstation. Baker does not teach or suggest that this application object is utilized as an "agent" per the plain meaning of this term as understood in the art. Please refer to Sub-Heading I above for a definition of the term "agent."

As another example, claim 17 recites: "the web-server having the agent that includes a public key of the application server." The Office Action cites Baker (Col. 9, lines 10-12) as teaching this recitation. Specifically, Baker teaches:

> The preferred embodiment further associates a given HTTPS request with a logical session which is initiated and tracked by a "cookie jar server" 28 to generate a "cookie" which is a unique server-generated key that is sent to the client along with each reply to a HTTPS request. The client holds the cookie and returns it to the server as part of each subsequent HTTPS request. (Col. 9, lines 7-13)

This section teaches a server 28 that generates a "cookie" that is sent to a client. First, as discussed herein in Sub-Heading II, a "cookie" is **not** a public key. Second, the bolded section of claim 17 recites four different elements: (1) a web-server, (2) an agent, (3) a public key, and (4) an application server. The portion of Baker cited by the Office Action does not even include four different elements. Appellants respectfully asked the Office Action to specify the portions of Baker that correspond with the elements of claim

17. The Office Action has not complied with this request. Appellants contest that Baker does not teach or suggest each of these four claimed elements.

Thus, the cited art does not teach or suggest each and every limitation of claim 17. All dependent claims that depend from independent claim 17 inherit all limitations of the base claim. For at least the reasons given in connection with claim 17, the dependent claims are also allowable over Baker.

Sub-Heading V: Claim 22 not anticipated by Baker

Claim 22 is rejected under 35 U.S.C. §102 as allegedly anticipated by Baker. Claim 22 recites numerous limitations that are not taught or suggested in Baker; examples are discussed below. For convenience, claim 22 is reproduced (emphasis added):

> A method, comprising:
>
> embedding in code of **an agent** a public key of an application server that is behind a firewall;
>
> downloading the code of **the agent** and the public key into a browser;
>
> verifying **the agent** to authenticate the public key of the application server;
>
> establishing a communication link between the application server and a relay module that is in a non-secure environment and between the browser and the relay module; and
>
> securely transferring data from the browser through the relay module to the application server without requiring a trusted intermediate party.

On numerous occasions, claim 22 recites recitations pertaining to an **agent**. Baker does not teach the limitations of claim 22 regarding the agent. For example, the Office Action cites Baker (at Col. 5, lines 60-61) for teaching an "agent." This section of Baker teaches a client tier of software on a customer workstation that has "one or more downloadable application objects directed to front-end business logic." Baker, thus, teaches downloading an application object into a workstation. Baker does not teach or suggest that this application object is utilized as an "agent" per the plain meaning of this

term as understood in the art. Please refer to Sub-Heading I above for a definition of the term "agent."

As yet numerous other examples, Appellants note at least the following occurrences of the term "agent" in claim 22:

1) embedding in code of **an agent** a public key of an application server,
2) downloading the code of **the agent**, and
3) verifying **the agent**.

Baker does not teach or suggest any of these limitations. Instead, the Office Action is constructing a "piecemeal examination" of the art and claims. For example, claim 22 recites "embedding in code of an agent a public key of an application server." The Office Action cites a section of Baker (Col. 9, lines 10-12) that discusses "cookies." As discussed herein in Sub-Heading II, a "cookie" is **not** a public key. These two terms have entirely different meanings to one of ordinary skill in the art (see citations herein to Webopedia).

As yet another example, claim 22 recites "verifying the agent." The Final Office Action does not even address this recitation (see Final OA, pages 9-10). In other words, the Final Office Action does not provide a section in Baker that teaches or suggests this recitation. Appellants have reviewed Baker and find no such teaching or suggestion.

Thus, the cited art does not teach or suggest each and every limitation of claim 22. All dependent claims that depend from independent claim 22 inherit all limitations of the base claim. For at least the reasons given in connection with claim 22, the dependent claims are also allowable over Baker.

III. Claims Rejection (Claims 6-8) – 35 USC § 103(a)

Claims 6-8 are rejected under 35 USC §103(a) as being unpatentable over Baker in view of Curry et al. (USPN 6,237,095). Claims 6-8 depend from claim 1 and, hence, inherit all the limitations of the base claim. Since Curry does not cure the deficiencies of Baker, claims 6-8 are allowable over the combination of Baker and Curry.

IV. Claims Rejection (Claims 9-10) – 35 USC § 103(a)

Claims 9-10 are rejected under 35 USC §103(a) as being unpatentable over Baker in view of Boyle et al. (USPN 6,119,167). Claims 9-10 depend from claim 1 and, hence, inherit all the limitations of the base claim. Since Boyle does not cure the deficiencies of Baker, claims 9-10 are allowable over the combination of Baker and Boyle.

V. Claims Rejection (Claims 12 and 16) – 35 USC § 103(a)

Claim 12 is rejected under 35 USC §103(a) as being unpatentable over Baker in view of Bradley et al. (USPN 6,584,507). Claim 12 depends from claim 11 and, hence, inherits all the limitations of the base claim. Since Bradley does not cure the deficiencies of Baker, claim 12 is allowable over the combination of Baker and Bradley.

VI. Claim Rejection (Claim 13) – 35 USC § 103(a)

Claim 13 is rejected under 35 USC §103(a) as being unpatentable over Baker in view of Bradley and Curry. Claim 13 depends from claim 11 and, hence, inherits all the limitations of the base claim. Since Bradley and Curry do not cure the deficiencies of Baker, claim 13 is allowable over the combination of Baker, Bradley, and Curry.

VII. Claim Rejection (Claim 24) – 35 USC § 103(a)

Claim 24 is rejected under 35 USC §103(a) as being unpatentable over Baker in view of Curry. Claim 24 depends from claim 22 and, hence, inherits all the limitations of the base claim. Since Curry does not cure the deficiencies of Baker, claim 24 is allowable over the combination of Baker and Curry.

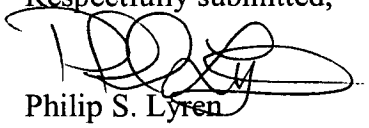
CONCLUSION

In view of the above, Appellants respectfully request the Board of Appeals to reverse the Examiner's rejection of all pending claims.

Any inquiry regarding this Appeal should be directed to Philip S. Lyren at Telephone No. (281) 514-8236, Facsimile No. (281) 514-8332. In addition, all correspondence should continue to be directed to the following address:

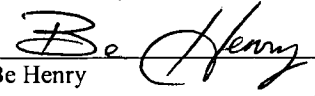
Hewlett-Packard Company
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400

Respectfully submitted,



Philip S. Lyren
Reg. No. 40,709
Ph: 281-514-8236

CERTIFICATE UNDER 37 C.F.R. 1.8: The undersigned hereby certifies that this paper or papers, as described herein, are being deposited in the United States Postal Service, as first class mail, in an envelope address to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on this 27th day of December, 2004.

By 
Name: Be Henry

Claims Appendix

1. (previously presented) A method for securely transferring data between an agent and an application server through a non-secure node comprising:

(a) establishing a session key between the agent and the application server by utilizing a public key of the application server; wherein the public key of the application server is embedded in the agent to enable the agent to derive the session key; and

(b) establishing an end-to-end secure connection between the agent and the application server by using the session key and by establishing a communication link between the application server and the non -secure node by using a relay module.

2. (previously presented) The method of claim 1 wherein establishing a communication link between the application server and the non-secure node by using a relay module comprises:

dynamically instantiating, by the application server, the relay module having a first port for communicating with the application server and a second port for communicating with the agent, the relay module listening on a first predetermined port number on the first port and a second predetermined port number on the second port; and

the application server connecting to the first port of the relay module to establish a connection therewith.

3. (original) The method of claim 2 wherein establishing a communication link between the application server and the agent through a relay module further comprises:

pushing data encrypted by the established session key from the agent to the application server over the end-to-end secure connection.

4. (original) The method of claim 2 wherein establishing a communication link between the application server and the agent through a relay module further comprises:

pulling data encrypted by the session key from the application server over the end-to-end secure connection to the agent.

5. (original) The method of claim 1 wherein establishing a session key between the agent and the application server by utilizing a public key of the application server further comprises:

establishing a shared secret key between the application server and the agent for encrypting and decrypting data sent therebetween.

6. (original) The method of claim 5 wherein establishing a shared secret key between the application server and the agent for encrypting and decrypting data sent therebetween comprises:

encrypting the shared secret key with the public key of the application server to generate an encrypted shared key;

sending the encrypted shared secret key to the application server; and

decrpyting the shared secret key with the private key of the application server.

7. (original) The method of claim 5 wherein establishing a shared secret key between the application server and the agent utilizes a key transfer protocol.

8. (original) The method of claim 7 wherein the key transfer protocol is the Rivest, Shamir, Adleman (RSA) public key algorithm.

9. (original) The method of claim 5 wherein establishing a shared secret key between the application server and the agent for encrypting and decrypting data sent therebetween utilizes a key agreement protocol.

10. (original) The method of claim 9 wherein the key agreement protocol is the Diffie-Hellman (DH) public key algorithm.

11. (previously presented) The method of securely transferring data between an application server and an agent of the application server through a non-secure environment having a web-server and the agent, the method comprising:

a) a user accessing the web-server to download the agent therefrom; wherein the agent includes a public key of the application server;

b) the agent deriving a shared session key with the application server by using the public key of the application server, the shared session key for use in encrypting and decrypting data to be transferred between the agent and the application server;

c) the application server establishing a connection to the web-server; and

d) the agent contacting the web server by using a first protocol to send data encrypted by the session key to the application server over the connection between the web-server and the application server.

12. (original) The method of claim 11 wherein the application server establishing a connection to the web-server further comprises

c1) the application server dynamically instantiating a relay module by sending a URL associated with the relay module to the web-server, the URL specifying a first predetermined port for communication between the web-server and the relay module;

c2) the application server connecting to the relay module on a first predetermined port; and

c3) the application server reading data from the relay module through the connection on the first predetermined port.

13. (original) The method of claim 12 wherein the agent contacting the web-server by using a first protocol to send data encrypted by the session key to the application server over the connection between the web-server and the application server further comprises

d1) the agent encrypting the session key with the public key of the application server;

d2) the agent collecting data;

d3) the agent encrypting the collected data with the session key;

d4) sending the encrypted session key and encrypted measured data to the application server by using a forwarding module that connects to a second predetermined port of the relay module.

14. (original) The method of claim 11 wherein the first protocol is one of HTTP and HTTP/SSL.

15. (previously presented) A secure data transfer system for connecting a non-secure node to an application server behind a firewall comprising:

a) a web-server in the non-secure node;

b) a relay in the non-secure node that is dynamically instantiated by the application server, the relay being configured by the application server to have a first port for listening for a connection from the application server;

wherein the application server connects to the relay on the first port and reads data from the first port.

16. (previously presented) The secure data transfer system of claim 15 wherein the relay does not initiate the connection with the application server but waits for the application server to establish the connection.

17. (original) A secure data transfer system for establishing an end-to-end secure connection between an agent and an application server behind a firewall through a non-secure node comprising:

a) a web-server residing in the non-secure node, the web-server having the agent that includes a public key of the application server;

b) a browser in communication with the web-server for downloading the agent from the web-server;

c) a secure transfer module residing in the non-secure node; and

d) an application server in a secure zone for initiating a connection to the web-server via the secure transfer module.

18. (original) The secure data transfer system of claim 17 wherein the secure transfer module further comprises:

c1) a relay module for listening to a first port and a second port;

c2) an instantiation module for executing the relay module in response to a command from the application server;

c3) a forwarding module for transferring data from the agent to the relay module in response to a command from the agent; and

wherein the relay module listens to the first port for a connection by the application server and listens to the second port for a connection by the forwarding module.

19. (original) The secure data transfer system of claim 16 wherein the non -secure node is a web-server node.

20. (previously presented) The method of claim 1 further comprising transferring data between the agent and the relay module via an unsecure communication link.

21. (previously presented) The method of claim 11 comprising transferring data between the agent and the web-server via an unsecure communication link.

22. (previously presented) A method, comprising:

embedding in code of an agent a public key of an application server that is behind a firewall;

downloading the code of the agent and the public key into a browser;

verifying the agent to authenticate the public key of the application server;

establishing a communication link between the application server and a relay module that is in a non-secure environment and between the browser and the relay module; and

securely transferring data from the browser through the relay module to the application server without requiring a trusted intermediate party.

23. (canceled)

24. (previously presented) The method of claim 22 further comprising collecting data with the agent.

25. (previously presented) The method of claim 24 wherein collecting data with the agent further comprises measuring time required to load data into the browser.

26. (previously presented) The method of claim 22 wherein the communication link between the browser and the relay module is an unsecure communication link.